UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/783,673	02/14/2001	Peter M. Mansour	SPROQ1100-1	9661

29585 7590 06/17/2005

DLA PIPER RUDNICK GRAY CARY US LLP
153 TOWNSEND STREET
SUITE 800
SAN FRANCISCO, CA 94107-1907

EXAMINER	
ZHONG, CHAD	

ART UNIT	PAPER NUMBER
2152	

DATE MAILED: 06/17/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/783,673	MANSOUR ET AL.	
	Examiner	Art Unit	
	Chad Zhong	2154	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *29 March 2005*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-53* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-53* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. This action is responsive to communications:

RCE Amendment, filed on 03/29/2005.

2. Claims 1-53 are presented for examination. In amendment A, filed on 03/29/2005:

Claims 1, 32, 36 are amended.

Applicant's remarks filed 03/29/2005 have been considered but are found not persuasive in view at the new grounds at rejection necessitated by Applicant's amendment.

Double Patenting

Claims 1-53 of this application conflict with claims 1-3, 5-6, 10-18, 46-47, 19-20, 29-31, 33-34, 36, 38, 45 of Application No. 09-783660. 37 CFR 1.78(b) provides that when two or more applications filed by the same applicant contain conflicting claims, elimination of such claims from all but one application may be required in the absence of good and sufficient reason for their retention during pendency in more than one application. Applicant is required to either cancel the conflicting claims from all but one application or maintain a clear line of demarcation between the applications. See MPEP § 822.

The nonstatutory double patenting rejection is based on a judicially created doctrine grounded in public policy (a policy reflected in the statute) so as to prevent the unjustified or improper timewise extension of the "right to exclude" granted by a patent and to prevent possible harassment by multiple assignees. See *In re Goodman*, 11 F.3d 1046, 29 USPQ2d 2010 (Fed. Cir. 1993); *In re Longi*, 759 F.2d 887, 225 USPQ 645 (Fed. Cir. 1985); *In re Van Ornum*, 686 F.2d 937, 214 USPQ 761 (CCPA 1982); *In re Vogel*, 422 F.2d 438, 164 USPQ 619 (CCPA 1970);and, *In re Thorington*, 418 F.2d 528, 163 USPQ 644 (CCPA 1969).

A timely filed terminal disclaimer in compliance with 37 CFR 1.321(c) may be used to overcome an actual or provisional rejection based on a nonstatutory double patenting ground provided the conflicting application or patent is shown to be commonly owned with this application. See 37 CFR 1.130(b).

Effective January 1, 1994, a registered attorney or agent of record may sign a terminal disclaimer. A terminal disclaimer signed by the assignee must fully comply with 37 CFR 3.73(b).

Claims 1-53 are rejected under the judicially created doctrine of obviousness-type double patenting as being unpatentable over claims 1-3, 5-6, 10-18, 46-47, 19-20, 29-31, 33-34, 36, 38, 45 of U.S. Patent No. 09-783660 in view of Simonoff et al. US 6,078,322.

Present Application: 09-783673	Co-Pending Application: 09-783660
1. A data processing method comprising: generating, with a client device, a particular form of a client-resident intermediate user interface (UI) for a server-based and client-side controlled application according to a UI format that is based **upon a number of device capabilities for said client device**, including supplementing a skeletal UI stored in a first memory location with one or more icons, labels or menu items, or combinations thereof, stored in a second memory location, wherein the first memory location and the second memory location are situated on said client device; **receiving, at said client device, a number of source data items related to said server-based application**; and populating at least one native UI control used by said intermediate U1 with said member of source data items.	1. A data processing method comprising: generating, with a client device, a particular form of a client-resident intermediate user interface (UI) for a server-based and client-side controlled application according to a U1 format determined by a UI server, including supplementing a skeletal UI stored in a first memory location with one or more icons, labels or menu items, or combinations thereof, stored in a second memory location. wherein the first memory location and the second memory location are situated on said client device; transmitting a number of source data items related to said server-based application from said U1 server to said client device; and populating at least one native U1 control used by said intermediate U1 with said number of source data items.

Current application does not explicitly teach "upon a number of device capabilities for said client device" and "receiving, at said client device, a number of source data items related to said server-based application", Simonoff teaches the above see for example, (Col. 7, lines 23-30; Col. 9, lines 33-50; Col. 11, lines 60-67) in order to partially updating the UI in steps in accordance with the GUIScript objects.

Present Application: 09-783673	Co-Pending Application: 09-783660
2. A method according to claim 1, wherein said at least one native UI control is associated with an operating system for said client device.	3. a method according to claim 1, wherein said at least one native UI control is associated with an operating system for said client device.

Present Application: 09-783673	Co-Pending Application: 09-783660
3. a method according to claim 1, further comprising the steps of: generating an action request in response to a manipulation of said intermediate UI by a user of said client device; and updating said intermediate UI in response to said action request.	5. a method according to claim 1, further comprising the steps of: generating an action request in response to a manipulation of said intermediate UI by a user of said client device; and updating said intermediate UI in response to said action request.

Present Application: 09-783673	Co-Pending Application: 09-783660
4. a method according to claim 1, further comprising the steps of: performing an offline action by said client device while said client device operates in a disconnected mode; subsequently establishing a session between said client device and a UI server; and thereafter transmitting, from said client device to said UI server, a command indicative of said offline action.	6. a method according to claim 1, further comprising the steps of: performing an offline action by said client device while said client device is disconnected from said UI server; subsequently establishing a session between said client device and said UI server; and thereafter transmitting, from said client device to said UI server, a command indicative of said offline action.

Present Application: 09-783673	Co-Pending Application: 09-783660
5. a method according to claim 1, further comprising the step of saving said number of source data items in a client cache resident at said client device.	10. a method according to claim 1, further comprising this step of saving said number of source data items in a client cache resident at said client device.

Present Application: 09-783673	Co-Pending Application: 09-783660
6. a method according to claim 5, further comprising the step of removing client cache items to accommodate said number of source data items.	11. a method according to claim 10, further comprising the step of removing client cache items to accommodate said number of source data items.

Present Application: 09-783673	Co-Pending Application: 09-783660
7. a method according to claim 6, wherein said removing step selectively removes said client cache items according to a hierarchical preference scheme.	12. a method according to claim 11, wherein said removing step selectively removes said client cache items according to a hierarchical preference scheme.

Present Application: 09-783673	Co-Pending Application: 09-783660
8. a method according to claim 1, further comprising the steps of: receiving, at said client device, a client action command related to said server-based application; and executing said client action command by said client device.	13. a method according to claim 1, further comprising the steps of: sending a client action command related to said server-based application from said UI server to said client device; and executing said client action command by said client device.

Present Application: 09-783673	Co-Pending Application: 09-783660
9. a method according to claim 1, wherein said number of source data items received during said receiving step represent a portion of a larger amount of related data available at a UI server.	14. a method according to claim 1, wherein said number of source data items represent a portion of a larger amount of related data available at said UI server.

Present Application: 09-783673	Co-Pending Application: 09-783660
10. (Original) A method according to claim 9 wherein: said larger amount of related data comprises a list of items; and said number of source data items represents a subset of said list of items.	15. a method according to claim 14, wherein: said larger amount of related data comprises a list of items; and said number of source data items represents a subset of said list of items.

Present Application: 09-783673	Co-Pending Application: 09-783660
11. a method according to claim 9, wherein: said larger amount of related data comprises a document; and said number of source data items represents a portion of said document.	16. a method according to claim 14, wherein: said larger amount of related data comprises a document; and said number of source data items represents a portion of said document.

Present Application: 09-783673	Co-Pending Application: 09-783660
12. a method according to claim 9, wherein: said larger amount of related data comprises an image; and said number of source data items represents a portion of said image.	17. a method according to claim 14, wherein: said larger amount of related data comprises an image; and said number of source data items represents a portion of said image.

Present Application: 09-783673	Co-Pending Application: 09-783660
13. a method according to claim 9, wherein: said larger amount of related data comprises a body of text; and said number of source data items represents a portion of said body of text.	18. a method according to claim 14, wherein: said larger amount of related data comprises a body of text; and said number of source data items represents a portion of said body of text.

Present Application: 09-783673	Co-Pending Application: 09-783660
14. a method according to claim 1, further comprising the step of retrieving a command script corresponding to a manipulation of a UI control contained in said intermediate UI, said command script being configured for execution by said client device	46. a method according to claim 45, further comprising the step of specifying a command script corresponding to a manipulation of a UI control contained in said UI form, said command script being configured for execution by said client device.

Present Application: 09-783673	Co-Pending Application: 09-783660
15. a method according to claim 14, further comprising the step of executing, by said client device, said command script in response to the manipulation of said UI control at said client device.	47. a method according to claim 46, further comprising the step of executing, by said client device, said command script in response to the manipulation of said UI control at said client device.

Present Application: 09-783673	Co-Pending Application: 09-783660
16. a method according to claim 15, wherein said executing step is performed by said client device in response to an offline manipulation of said U1 control at said client device.	6. a method according to claim 1, further comprising the steps of: performing an offline action by said client device while said client device is disconnected from said UI server; subsequently establishing a session between said client device and said UI server; and thereafter transmitting, from said client device to said UI server, a command indicative of said offline action.

Present Application: 09-783673	Co-Pending Application: 09-783660
17. a data processing method comprising: storing a user interface (UI) form definition locally at a client device, said U1 form definition being dictated by a number of device capabilities for said client device; said client device saving a number of source data items locally, said number of source data items being related to a served-based application; said client device rendering a UI that is based upon said UI form definition; and said client device populating said UI with said number of source data items, and wherein said number of source data items comprises a smaller subset than a total number of source data items related to said server-based application, and wherein further subsets of said total number of source data items are downloadable based upon execution of one or more client-side controls.	19. a data processing method comprising: defining a user interface (UI) form in response to a number of device capabilities for a client device; storing said UI form locally at said client device; saving a number of source data items locally at said client device, said number of source data items being related to a server-based application executed by a UI server; and populating said UI form with said number of source data items, and wherein said number of source data items comprises a smaller subset than a total number of source data items related to said server-based application, and wherein further subsets of said total number of source data items are downloadable based upon execution of one or more client-side controls.

Present Application: 09-783673	Co-Pending Application: 09-783660
18. a method according to claim 17, further comprising the step of receiving, at said client device, said number of source data items from a UI server.	20. a method according to claim 19, further comprising the step of transmitting said number of source data items from said UI server to said client device.

As per claims 19-22, claims 19-22 are rejected for the same reasons as rejection to claim 3-7 above.

Present Application: 09-783673	Co-Pending Application: 09-783660
23. a method according to claim 22, wherein said removing step selectively removes said client cache items according to a hierarchical preference scheme.	29. a method according to claim 28, wherein said removing step selectively removes said existing client cache items according to a hierarchical preference scheme.

Present Application: 09-783673	Co-Pending Application: 09-783660
24. a method according to claim 21, further comprising the steps of: updating said UI in response to a manipulation of a display control rendered by said client device; requesting an additional number of source data items if said manipulation of said display control triggers a data request command; and replacing source data items saved in said client cache with said additional number of source data items.	30. a method according to claim 27, further comprising the steps of: updating said UI form in response to a manipulation of a display control rendered by said client device; requesting an additional number of source data items from said U1 server if said manipulation of said display control triggers a data request command; and replacing source data items saved in said client cache with said additional number of source data items.

Present Application: 09-783673	Co-Pending Application: 09-783660
25. a method according to claim 21, further comprising the steps of: updating said UI in response to a manipulation of a display control rendered by said client device; retrieving additional source data items from said client cache in response to said manipulation of said display control; and displaying said additional source data items in said UI.	31. a method according to claim 27, further comprising the steps of: updating said UI form in response to a manipulation of a display control rendered by said client device; retrieving additional somce data items from said client cache in response to said manipulation of said display control; and displaying said additional source data items in said UI form.

As per claim 26, claim 26 is rejected for the same reasons as rejection to claim 8 above

Present Application: 09-783673	Co-Pending Application: 09-783660
27. a method according to claim 17, wherein said UI form definition is dictated by said served-based application.	33. a method according to claim 19, wherein said defining step defines said UI form based upon said server-based application.

Present Application: 09-783673	Co-Pending Application: 09-783660
28. a method according to claim 17, wherein said UI form definition identifies at least one native UI control stored locally at said client device.	34. a method according to claim 19, wherein said defining step defines said UI form with at least one native UI control stored locally at said client device.

As per claim 29, claim 29 is rejected for the same reasons as rejection to claim 9 above.

Present Application: 09-783673	Co-Pending Application: 09-783660
30. a method according to claim 29, further comprising the steps of: said client device generating a request for additional source data items; and said client device receiving, from said UI server, a subsequent portion of said total number of source data items.	36. a method according to claim 35, further comprising the steps of: said UI server receiving a request for additional source data items', and said UI server transmitting a subsequent portion of said total number of source data items to said client device in response to said request.

As per claim 31, claim 31 is rejected for the same reasons as rejection to claim 15 above.

Present Application: 09-783673	Co-Pending Application: 09-783660
32. a data processing method comprising: obtaining a user interface (UI) form definition for a served-based application, where said UI form definition is based upon a number of device capabilities for a client device; said client device receiving an instruction to render a particular UI form of a client-resident intermediate UI corresponding to said UI form definition; said client device rendering said particular UI form with at least one native UI control associated with an operating system for said client device, including supplementing a skeletal UI stored in a first memory location with one or more icons, labels or menu items, or combinations thereof, stored in a second memory location wherein the first memory location and the second memory location are situated on said client device; said client device obtaining a number of data items related to said server-based application; and said client device displaying said number of data items in said at least one native UI control.	38. a data processing method comprising: executing, at a user interface (UI) server, a server-based application configured to manipulate source data items for presentment at a client device; displaying a particular UI form of a client-resident intermediate UI at said client device according to a UI format determined by a UI server, including supplementing a skeletal U1 stored in a first memory location with one or more icons, labels or menu items, or combinations thereof stored in a second memory location, said U1 form being capable of presenting data items to a user of said client device. wherein the first memory location and the second memory location are situated on said client device; generating a client-side controlled action request in response to a manipulation of said UI form by a user of said client device; and updating said U1 form in response to said action request.

As per claim 33, claim 33 is rejected for the same reasons as rejection to claim 5 above.

As per claim 34, claim 34 is rejected for the same reasons as rejection to claim 25 above.

As per claim 35, claim 35 is rejected for the same reasons as rejection to claim 24 above.

Present Application: 09-783673	Co-Pending Application: 09-783660
36. (Currently Amended) A client device architecture for use with a client device capable of communicating with a data processing server, said client device architecture comprising: a receive module configured to receive an instruction that identifies a user interface (UI) form definition; an operating system; a number of native UI controls provided by said operating system; a UI form data cache configured to store said U1 form definition; and a UI module configured to generate a particular U1 form of a client-resident intermediate UI for a served-based application according to said U1 form definition, including supplementing a skeletal UI stored in a first memory location with one or more icons, labels or menu items, or combinations thereof, stored in a second memory location wherein the first memory location and the second memory location are situated on said client device. and to populate at least one of said native UI controls with a number of source data items associated with said server-based application.	38. a data processing method comprising: executing, at a user interface (UI) server, a server-based application configured to manipulate source data items for presentment at a client device; displaying a particular UI form of a client-resident intermediate UI at said client device according to a UI format determined by a UI server, including supplementing a skeletal U1 stored in a first memory location with one or more icons, labels or menu items, or combinations thereof stored in a second memory location, said U1 form being capable of presenting data items to a user of said client device. wherein the first memory location and the second memory location are situated on said client device; generating a client-side controlled action request in response to a manipulation of said UI form by a user of said client device; and updating said U1 form in response to said action request.

As per claims 37-39, claims 37-39 are rejected for the same reasons as rejection to claims 5-7 above.

As per claims 40-41, claims 40-41 are rejected for the same reasons as rejection to claims 24-25 above respectively.

Present Application: 09-783673	Co-Pending Application: 09-783660
42. a client device architecture according to claim 36, wherein said receive module is further configured to receive said number of source data items from a remote UI Server.	45. a data processing method comprising: generating a user interface (UI) form definition for a server-based application based upon a number of device capabilities for a client device; instructing said client device to render a UI form corresponding to said UI form definition; rendering said U1 form with at least one native

	UI control associated with an operating system for said client device; transmitting a number of data items from a UI server to said client device, said number of data items being related to said server-based application; and displaying said number of data items in said at least one native U1 control, and wherein said number of source data items comprises a smaller subset than a total number of source data items related to said server-based application, and wherein further subsets of said total number of source data items are downloadable based upon execution of one or more client-side controls.

Present Application: 09-783673	Co-Pending Application: 09-783660
43. a client device architecture according to claim 36, wherein said receive module is further configured to receive said UI form definition from a remote UI server.	45. a data processing method comprising: generating a user interface (UI) form definition for a server-based application based upon a number of device capabilities for a client device; instructing said client device to render a UI form corresponding to said UI form definition; rendering said U1 form with at least one native UI control associated with an operating system for said client device; transmitting a number of data items from a UI server to said client device, said number of data items being related to said server-based application; and displaying said number of data items in said at least one native U1 control, and wherein said number of source data items comprises a smaller subset than a total number of source data items related to said server-based application, and wherein further subsets of said total number of source data items are downloadable based upon execution of one or more client-side controls.

Present Application: 09-783673	Co-Pending Application: 09-783660
44. a client device architecture according to claim 36, wherein said UI form definition is based upon a number of device capabilities for said client device.	2. a method according to claim 1, further comprising the step of formatting characteristics of said intermediate UI based upon a number of device capabilities for said client device.

As per claims 45-47, 49, claims 45-47, 49 are rejected for the same reasons as rejection to claims 1, 5-6, 24 above respectively.

As per claim 50, claim 50 is rejected for the same reasons as rejection to combination of claims 24, 32 above.

As per claims 51-53, claims 51-53 are rejected for the same reasons as rejection to claims 29, 27, 32 above respectively.

As per claim 48, claim 48 is rejected for the same reasons as rejection to claim 7 above.

Claim Rejections - 35 USC § 102

3. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless --
>
> (e) the invention was described in a patent granted on an application for patent by another filed in the United States before the invention thereof by the applicant for patent, or on an international application by another who has fulfilled the requirements of paragraphs (1), (2), and (4) of section 371 (c) of this title before the invention thereof by the applicant for patent.

5. Claims 1-3, 5-6, 7, 8-12, 14-19, 21-22, 23, 24-47, 48, 49-53 are rejected under 35 U.S.C. 102(e) as being anticipated by Simonoff et al. (hereinafter Simonoff), US 6,078,322.

6. As per claim 1, Simonoff teaches a data processing method comprising:

generating, with a client device (universal client), a particular form of a client-resident intermediate user interface (UI) for a server-based and client-side controlled application according to a UI format that is based upon a number of device capabilities of said client device,

including supplementing a skeletal UI stored in a first memory location with one or more icons, labels

or menu items, or combinations thereof, stored in a second memory location (Col. 7, lines 23-30; Col. 9, lines 33-50; Col. 11, lines 60-67), wherein the first memory location and the second memory location are situated on said client device (Col. 14, lines 40-67, wherein everything can be implemented on a single client);

receiving, at said client device, a number of source data items related to said server-based application (Col. 9, lines 33-50); and

populating at least one native UI control used by said UI with said number of source data items (Col. 9, lines 33-50; Col. 14, lines 33-41; Col. 16, lines 40-49).

7. As per claim 2, Simonoff teaches a method according to claim 1, wherein said at least one native UI control is associated with an operating system for said client device (Col.11 lines 64-67).

8. As per claim 3, Simonoff teaches a method according to claim 1, further comprising the steps of:

generating an action request in response to a manipulation of said UI by a user of said intermediate client device (Col. 12, lines 1-13); and

updating said intermediate UI in response to said action request (Col. 12, lines 1-13).

9. As per claim 5, Simonoff teaches a method according to claim 1, further comprising the step of saving said number of source data items in a client cache resident at said client device (Col. 14, lines 44-56).

10. As per claim 6, Simonoff teaches a method according to claim 5, further comprising the step of removing client cache items to accommodate said number of source data items (Col. 13, lines 35-44; Col. 14, lines 44-56).

11. As per claim 8, Simonoff teaches a method according to claim 1, further comprising the steps of: receiving, at said client device, a client action command related to said server-based application; and executing said client action command by said client device (Col. 9, lines 33-50).

12. As per claim 9, Simonoff teaches a method according to claim 1, wherein said number of source data items received during said receiving step represent a portion of a larger amount of related data available at a UI server (Col. 16, lines 40-49; Col. 14, lines 44-56; Col 9, lines 33-50; Col. 10, lines 23-30).

15. As per claim 10, Simonoff teaches a method according to claim 9, wherein:

said larger amount of related data comprises a list of items; and said number of source data items represents a subset of said list of items (Col. 16, lines 40-49)

16. As per claim 11, Simonoff teaches a method according to claim 9, wherein:

said larger amount of related data comprises a document (Col. 9, lines 33-50); and

said number of source data items represents a portion of said document (Col. 9, lines 33-50; Col. 16, lines 40-49).

17. As per claim 12, Simonoff teaches a method according to claim 9, wherein:

said larger amount of related data comprises an image; and

said number of source data items represents a portion of said image (Col. 16, lines 40-49).

18. As per claim 14, Simonoff teaches a method according to claim 1, further comprising the step of retrieving a command script corresponding to a manipulation of a UI control contained in said intermediate UI, said command script being configured for execution by said client device (Col. 12, lines 1-13).

19. As per claim 15, Simonoff teaches a method according to claim 14, further comprising the step of executing, by said client device, said command script in response to the manipulation of said UI control at said client device (Col. 12, lines 1-13; Col. 10, lines 23-30).

20. As per claim 16, Simonoff teaches a method according to claim 15, wherein said executing step is performed by said client device in response to an offline manipulation of said UI control at said client device (Col. 10, lines 23-30).

21. As per claim 17, Simonoff teaches a data processing method comprising:

storing a user interface (UI) form definition locally at a client device, said U1 form definition being dictated by a number of device capabilities for said client device (Col. 10, lines 23-30);

said client device saving a number of source data items locally, said number of source data items being related to a served-based application (Col. 10, lines 23-30, lines 34-48; Col. 9, lines 33-50);

said client device rendering a UI that is based upon said UI form definition (Col. 9, lines 33-50); and

said client device populating said UI with said number of source data items, and wherein said number of source data items comprises a smaller subset than a total number of source data items related to said server-based application, and wherein further subsets of said total number of source data items are downloadable based upon execution of one or more client-side controls (Col. 9, lines 33-50).

22. As per claim 18, Simonoff teaches a method according to claim 17, further comprising the step of receiving, at said client device, said number of source data items from a UI server (Col. 9, lines 33-50).

23. As per claims 19, 21-22, claims 19, 21-22 are rejected for the same reasons as rejection to claims 3, 5-7 above.

24. As per claim 24, Simonoff teaches a method according to claim 21, further comprising the steps of:

updating said UI in response to a manipulation of a display control rendered by said client device (Col. 12, lines 1-13);

requesting an additional number of source data items if said manipulation of said display control triggers a data request command (Col. 12, lines 1-13); and

replacing source data items saved in said client cache with said additional number of source data items (Col. 13, lines 35-44).

25. As per claim 25, Simonoff teaches a method according to claim 21, further comprising the steps of:

updating said UI in response to a manipulation of a display control rendered by said client device (Col. 12, lines 1-13);

retrieving additional source data items from said client cache in response to said manipulation of said display control (Col. 12, lines 1-13; Col. 10, lines 23-30); and

displaying said additional source data items in said UI (Col. 12, lines 1-13).

26. As per claim 26, claim 26 is rejected for the same reasons as rejection to claim 8 above.

27. As per claim 27, Simonoff teaches a method according to claim 17, wherein said UI form definition is dictated by said server-based application (Col. 7, lines 22-30; Col. 9, lines 33-50).

28. As per claim 28, Simonoff teaches a method according to claim 17, wherein said UI form definition identifies at least one native UI control stored locally at said client device (Col. 10, lines 23-30).

29. As per claim 29, claim 29 is rejected for the same reasons as rejection to claim 9 above.

30. As per claim 30, Simonoff teaches a method according to claim 29, further comprising the steps of:

said client device generating a request for additional source data items; and

said client device receiving, from said UI server, a subsequent portion of said total number of source data items (Col. 12, lines 1-13; Col. 16, lines 40-49; Col. 14, lines 44-56; Col. 9, lines 33-50; Col. 10, lines 23-30).

31. As per claim 31, claim 29 is rejected for the same reasons as rejection to claim 15 above.

32. As per claim 32, Simonoff teaches a data processing method comprising:

obtaining a user interface (UI) form definition for a served-based application, where said UI form definition is based upon a number of device capabilities for a client device (Col. 9, lines 33-50);

said client device receiving an instruction to render a particular UI form of a client-resident intermediate UI corresponding to said UI form definition (Col. 9, lines 33-50);

said client device rendering said particular UI form with at least one native UI control associated with an operating system for said client device, including supplementing a skeletal UI stored in a first memory location with one or more icons, labels or menu items, or combinations thereof, stored in a second memory location (Col. 11, lines 64-67) wherein the first memory location and the second memory location are situated on said client device (Col. 14, lines 40-67);

said client device obtaining a number of data items related to said server-based application (Col. 9, lines 33-50); and

said client device displaying said number of data items in said at least one native UI control (Col. 9, lines 33-50; Col. 10, lines 8-10; Col. 12, lines 59-62).

33. As per claim 33, claim 33 is rejected for the same reasons as rejection to claim 5 above.

34. As per claim 34, claim 34 is rejected for the same reasons as rejection to claim 25 above.

35. As per claim 35, claim 35 is rejected for the same reasons as rejection to claim 24 above.

36. As per claim 36, Simonoff teaches a client device architecture for use with a client device capable of communicating with a data processing server, said client device architecture comprising:

a receive module configured to receive an instruction that identifies a user interface (UI) form definition (Col. 9, lines 33-50);

an operating system (Col. 11, lines 64-67);

a number of native UI controls provided by said operating system (Col. 10, lines 23-30);

a UI form data cache configured to store said U1 form definition (Col. 10, lines 23-30; Col. 8, lines 15-19); and

a UI module configured to generate a particular U1 form of a client-resident intermediate UI for a served-based application according to said U1 form definition (Col. 9, lines 33-50), including supplementing a skeletal UI stored in a first memory location with one or more icons, labels or menu items, or combinations thereof, stored in a second memory location wherein the first memory location and the second memory location are situated on said client device (Col. 14, lines 40-67); and

to populate at least one of said native UI controls with a number of source data items associated with said server-based application (Col. 9, lines 33-50).

37. As per claim 37-39, claims 37-39 are rejected for the same reasons as rejection to claim 5-7 above.

38. As per claim 40-41, claims 40-41 are rejected for the same reasons as rejection to claim 24-25

above.

39. As per claim 42, Simonoff teaches a client device architecture according to claim 36, wherein said receive module is further configured to receive said number of source data items from a remote UI server (Col. 9, lines 33-50).

40. As per claim 43, Simonoff teaches a client device architecture according to claim 36, wherein said receive module is further configured to receive said UI form definition from a remote UI server (Col. 9, lines 33-50).

41. As per claim 44, Simonoff teaches a client device architecture according to claim 36, wherein said UI form definition is based upon a number of device capabilities for said client device (Col. 9, lines 33-50; Col. 11, lines 64-67).

42. As per claims 45-47, 49, claims 45-47, 49 are rejected for the same reasons as rejection to claim 1, 5-6, 24 above respectively.

43. As per claim 50, claim 50 is rejected for the same reasons as rejection to combination of claims 24 and 32 above.

44. As per claims 51-53, claims 51-53 are rejected for the same reasons as rejection to claim 29, 27 and 32 above respectively.

45. As per claim 7, Simonoff teaches a method according to claim 6, wherein said removing step selectively removes said client cache items according to a hierarchical preference scheme (Col. 13, lines 35-52).

46. As per claim 23, Claim 23 is rejected for the same reasons as rejection to claim 7 above.

47. As per claim 48, claim 48 is rejected for the same reasons as rejection to claim 7 above.

Claim Rejections - 35 USC # 103

48. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

49. Claim 13 is rejected under 35 U.S.C. 103(a) as being unpatentable over Simonoff et al. (hereinafter Simonoff, US 6,078,322, in view of 'Official Notice'.

50. As per claim 13, Simonoff does not teach a method according to claim 9, wherein:

said larger amount of related data comprises a body of text; and

said number of source data items represents a portion of said body of text. However 'Official Notice' is taken by the Examiner that a text file is notoriously well known as a type of file. It would have been obvious to have used a text file for the purpose of the current invention, because doing so would be less burdening for the individual units, through the usage of text file in place of image or a document and the like, the user now have the option of manipulating a portion of the text file thereby improving processing efficiency and speed on the client side, thus realizing a thin client network.

51. Claims 4, 20 are rejected under 35 U.S.C. 103(a) as being unpatentable over Simonoff et al. (hereinafter Simonoff, US 6,078,322, in view of "Browser wars: Rest in peace", Patrick, January 2001.

52. As per claim 4, Simonoff teaches a method according to claim 1, further comprising the steps of:

performing an offline action by said client device while said client device operates in a disconnected

mode (Col. 10, lines 23-30);

subsequently establishing a session between said client device and a UI server (Col. 17, lines 10-15; wherein the client establishes with server if there is a connection failure or switch over to a back up server);

Simonoff does not explicitly teaches:

thereafter transmitting, from said client device to said UI server, a command indicative of said offline action.

Patrick teaches of a system wherein the users has the option of manipulating data while offline and the changed data gets updated when the software goes online again, in order to allow for an offline usage option for the users.

Patrick teaches:

thereafter transmitting, from said client device to said UI server, a command indicative of said offline action (pg 2, 2nd paragraph).

It would have been obvious to combine teachings of Simonoff and Patrick in order to provide the user to an offline manipulate and access of data (pg 1, 10th paragraph).

53. As per claim 20, claim 20 is rejected for the same reasons as rejection to claim 4 above.

Conclusion

54. In the remark, the Applicant argued in substance that Simonoff fails to disclose or suggest "indication in the reference that the GUIScript file or the GUI is anything less than an entire UI displayed to a user... GUI objects as being skeletal nor supplementing the same"

In response to Applicant's arguments, Simonoff teaches the above limitations. See for example, Col. 12, lines 24-60, wherein the GUIScripts carries a GUI change in response to an event, the event, i.e. button

clicking, has resulted in partial GUI updating, only a portion of the GUI are generated at a time, any additional changes applies to the existing GUI are based upon system events, thus, GUI elements are supplemented in accordance with client requests, (see also, Col. 11, lines 55-67; Col. 12, lines 40-45). Hence, Simonoff teaches the foregoing section.

55. In the remark, the Applicant argued in substance that Simonoff realizes the use of two memory locations only be pushing an entire GUI object "from the server to the client", however, Simonoff fails to disclose or suggest "both memory locations to be situated on the client device".

In response to Applicant's arguments, the Examiner agrees with the applicant that pushing GUI objects from server to client is taught by Simonoff. However, in another embodiment, Simonoff discloses of a system running as a stand alone architecture and carry out similar methodologies as the client server model, exchanging GUI objects, see for example, Col. 14, lines 33-67.

Hence, Simonoff teaches the newly amended section as presented above.

56. In the remark, the Applicant argued in substance that Simonoff fails to disclose or suggest "GUIScript populating a native UI control".

In response to Applicant's arguments, Simonoff teaches this limitation. See for example Fig 7, in a non-limiting example, Simonoff discloses of Chatroom GUI, in a chatroom, the GUI is updated when the user selects the event button, i.e. send/clear/close, send will display the new messages to users of the chatroom, thus only the portion of the GUI gets updated periodically in accordance to user actions, while remainder of the GUI remains fixed, native UI control is thus populated with user entries. Furthermore, in Col. 12, lines 25-67, the GUI gets an update upon a system event. The 'send' event would cause partial update of the GUI as shown in Fig 7, thus realizing GUIScript populating a native UI control.

57. In the remark, Applicant argued in substance that portions of claim 4 and 20 are missing, please refer to the action above for additional details regarding those claims.

58. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure. The following patents and publications are cited to further show the state of the art with respect to "PLATFORM-INDEPENDENT DISTRIBUTED USER INTERFACE CLIENT ARCHITECTURE".

i. US 5818447 Wolf et al.

ii. US 2002/0152244 Dean et al.

iii. US 6167534 Straathof et al.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Chad Zhong whose telephone number is (571)272-3946. The examiner can normally be reached on M-F 7:15 to 4:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, BURGESS, GLENTON B can be reached on (571)272-3949. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

CZ
May 2, 2005

GLENTON B. BURGESS
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100